


# 二叉平衡树的旋转操作

原创 取酌由君 于 2018-06-25 00:33:21 发布 26413 收藏 86 版权

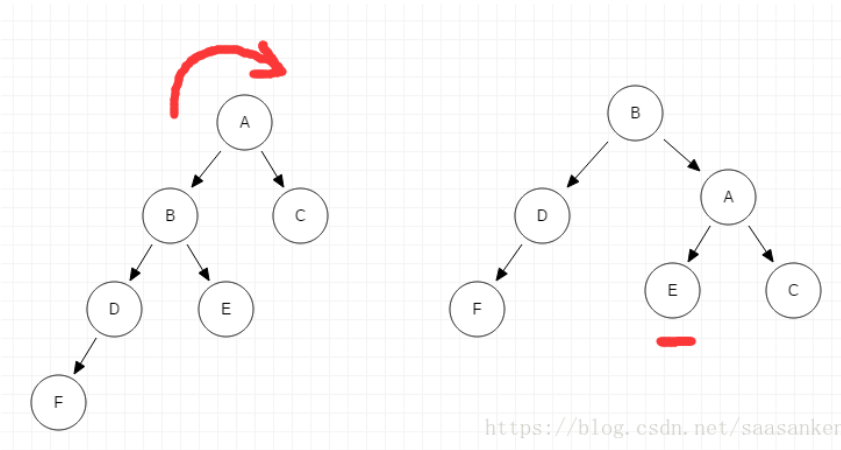
分类专栏: 数据结构 算法 文章标签: 数据结构 算法

 数据结构 同时被 2 个专栏收录

0 订阅 1 篇文章 订阅专栏

旋转是很多二叉平衡树维持平衡的主要手段，在这里复习一下。其实旋转过程中节点位置的变化只要遵循一个原则就行了：比Root小的在左子树，比Root大的在右子树。（当然这里前提条件是左小右大）。

情况一：插入F节点导致失衡：

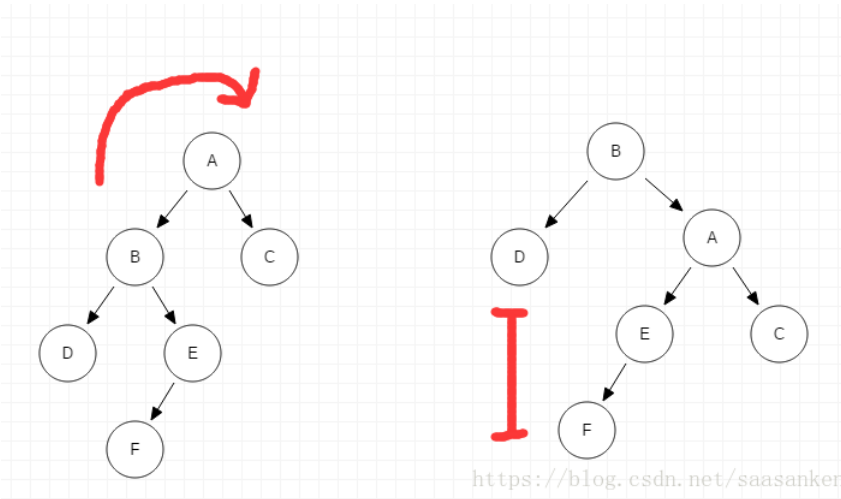


这里失衡的是A的左右子树，很容易就可以想到旋转B-A链，值得注意的是E节点，它原先在B的右子树，现在也依然在B的右子树，它原先在A的左子树，现在也依然在A的左子树。

若插入F节点在D的右子树处，旋转操作依然是上图那样，不谈。


但如果插入F节点是E的孩子就不一样了。

情况二：插入F节点导致失衡：



插入节点是E的孩子时，如果我们还像上面那样旋转B-A链，旋转后的树依然是不平衡的。事实上，这样的旋转使得B成为了新的根节点，而原图中比B大的节点有4个，比B小的节点只有D，若B为根其左子树只能为D，必定是不平衡的。

我们仔细观察原图，这里E节点是很特殊的节点。首先它是实际执行了插入操作的节点，其次图中比E小的节点有B、D、F，比E大的节点有A、C。如果我们能够让E节点做新的根节点就很好平衡了，那么怎样让E节点“上位”呢？

 取酌由君

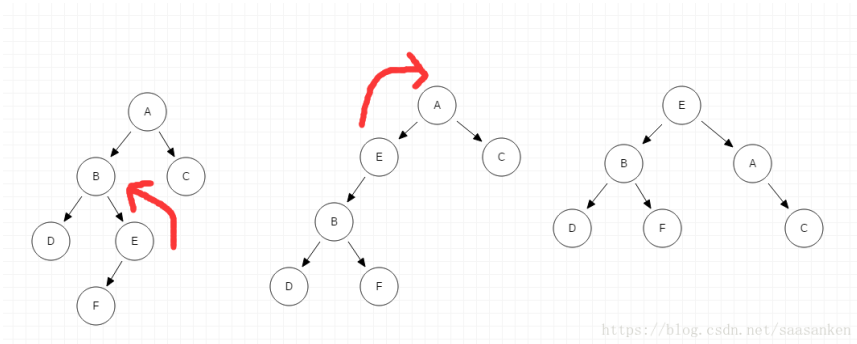
关注

66

26

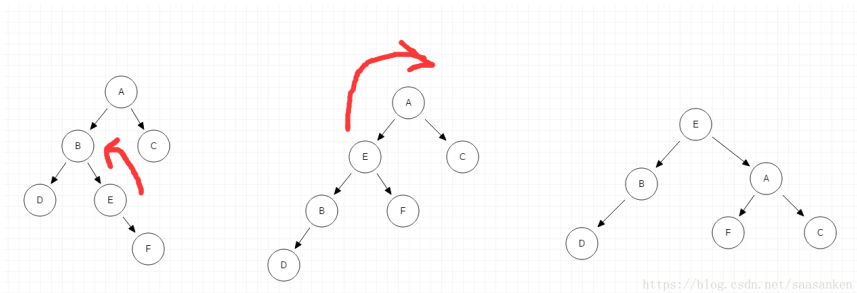
86

方法是进行两次旋转，如下图：



E节点恐成最大赢家.....

插入节点是E的右子树的情况与之类似，这里给出旋转图：

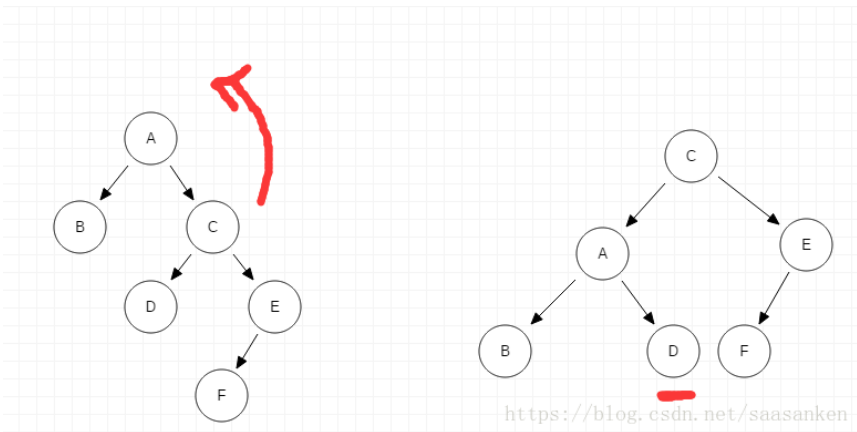


F比E大，双旋之后还是在E的右节点。

双旋看图理解起来简单，实际实现时要注意，我们可以判断失衡的是A节点，由A有直接关系的是B和C，那么我们怎么知道新插入的F节点是D的子树还是E的子树呢？这里的方法是比较F节点值与B的大小，大则是E的子树，要左-右共两次旋转，小则是D的子树，要一次右旋转。

当然还要考虑镜像情况：

情况三：插入 F 导致失衡：



进行一次左旋，关注D节点，它比C小，旋转后依然在C的左子树。

情况四：插入F导致失衡：



取酌由君

关注

66

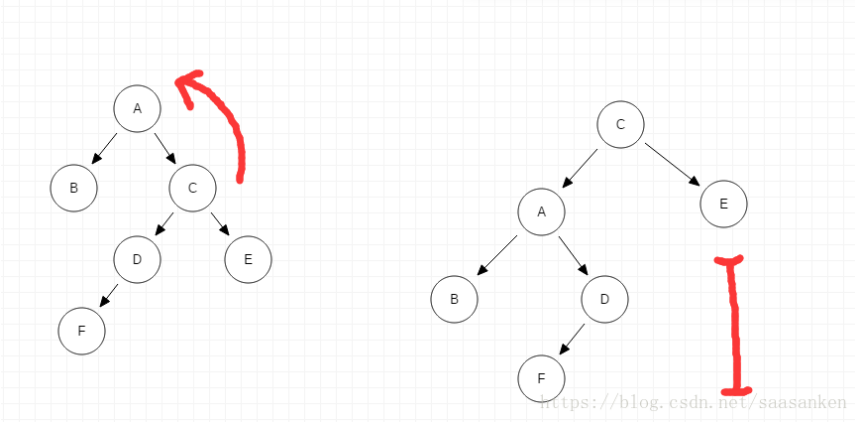


26

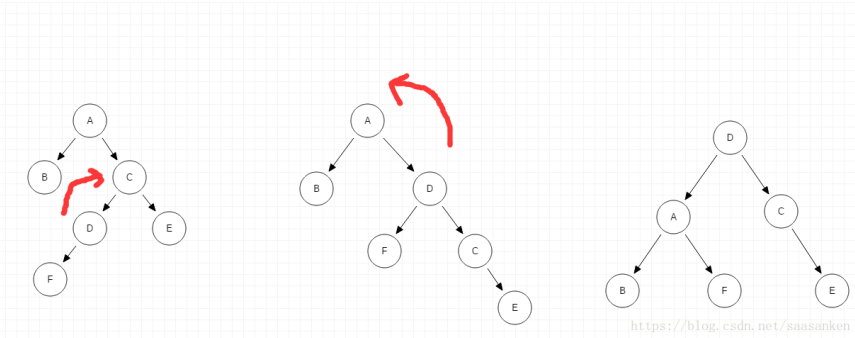


86



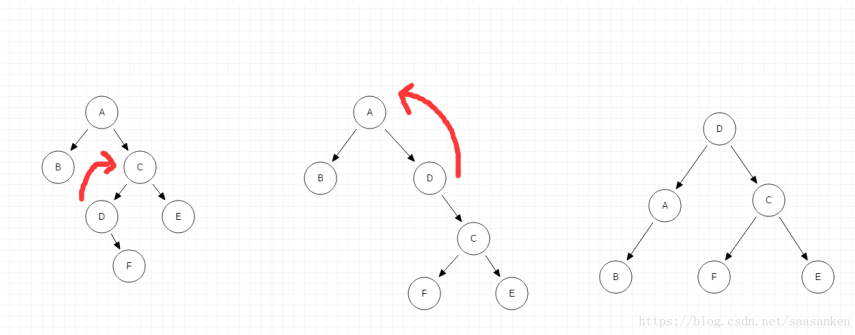


这时候左旋失败，理由和之前右旋失败类似，比C节点大的节点只有一个E，C是没法做新的根节点的。这里特殊的是D节点。



进行右-左两次双旋就可以了。D节点：爽到.....

插入节点是D的右子树情况类似，RT：



平衡二叉树的左右旋以及双旋转的图文详解 08-26

今天小编就为大家分享一篇关于平衡二叉树的左右旋以及双旋转的图文详解，小编觉得内容挺不错的...

平衡二叉树旋转平衡（要看）！ kllihuang的专栏 4099

这个恐怕是整个《数据结构》教科书里面最难的和最“没用”的数据结构了（现在的教科书还有部分算法...

评论 26 您还未登录，请先 登录 后发表或查看评论

...实现原理(Java实现)\_F\_chords的博客\_平衡二叉树旋转... 3-12

我们把目光放在“1”这个节点上，1的左子树高度为0，但是右子树高度为2，其差值为2大于1，所以这不是一...

平衡二叉树（树的旋转） 蓝天的博客 1385

1.概念 平衡二叉树建立在二叉排序树的基础上，目的是使二叉排序树的平均查找长度更小，即让各结...

平衡树（旋转） 最新发布 qq\_44866969的博客 19

#include<bits/stdc++.h> using namespace std; struct node{ int val; node\* left; node\* right; }; node\* rot...

平衡二叉树的旋转以及简便方法 hello\_program\_world的博客 678

刚开始听这个平衡二叉树的旋转，一听就蒙了，后来看了很多视频，有很多的说法。下面来介绍平衡...

AVL平衡二叉树中旋转操作的本质及其实现 热门拼 取酌由君 关注

66 26 86